

Sequenciamento em uma máquina minimizando o atraso

utilizando Algoritmos Genéticos

**Claudio Busatto
Guilherme Severo**

Definição

- * n processos devem ser processados sem interrupção em uma máquina.
- * cada processo possui um tempo de processamento, um peso e um prazo.
- * um processo é completo em um tempo C_j .
- * o seu atraso é $T_j = \max(C_j - d_j, 0)$.
- * o problema consiste em encontrar uma ordem de processamento com atraso total ponderado mínimo.

Exemplo

O conjunto de processos abaixo, possui uma solução ótima 2-1-4-5-3-6-7, com um atraso total de 454.

j	1	2	3	4	5	6	7
<i>p_j</i>	12	13	14	16	26	31	32
<i>d_j</i>	42	33	51	48	63	88	146
<i>w_j</i>	7	9	5	14	10	11	8

Formulação

$$\min \sum_{i \in N} w_i T_i$$

s.a.

$$T_i \geq C_i - d_i \quad \forall i \in N \quad (\text{R1})$$

$$T_i \geq 0 \quad \forall i \in N \quad (\text{R2})$$

$$\sum_{i \in N \cup \{0\}, i \neq j} x_{ij} = 1, \quad \forall j \in N \cup \{0\} \quad (\text{S1})$$

$$\sum_{j \in N \cup \{0\}, j \neq i} x_{ij} = 1, \quad \forall i \in N \cup \{0\} \quad (\text{S2})$$

$$C_j \geq C_i - M + (p_j + M)x_{ij} \quad \forall i \in N \cup \{0\}, \forall j \in N \quad (\text{S3})$$

$$C_i \geq 0 \quad \forall i \in N, C_o = 0 \quad (\text{S4})$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in N \cup \{0\}, \forall j \in N \cup \{0\} \quad (\text{S5})$$

Resultados GLPK

Instância	Melhor valor conhecido	Valor obtido no GLPK após 4h
Wt100/1	5988	58713
Wt100/2	6170	66092
Wt100/3	4267	51986
Wt100/5	5283	43650
Wt100/7	50972	197790
Wt100/11	181649	429880
Wt100/13	178840	473290
Wt100/17	332804	692306
Wt100/19	477684	867564
Wt100/23	539716	1097539

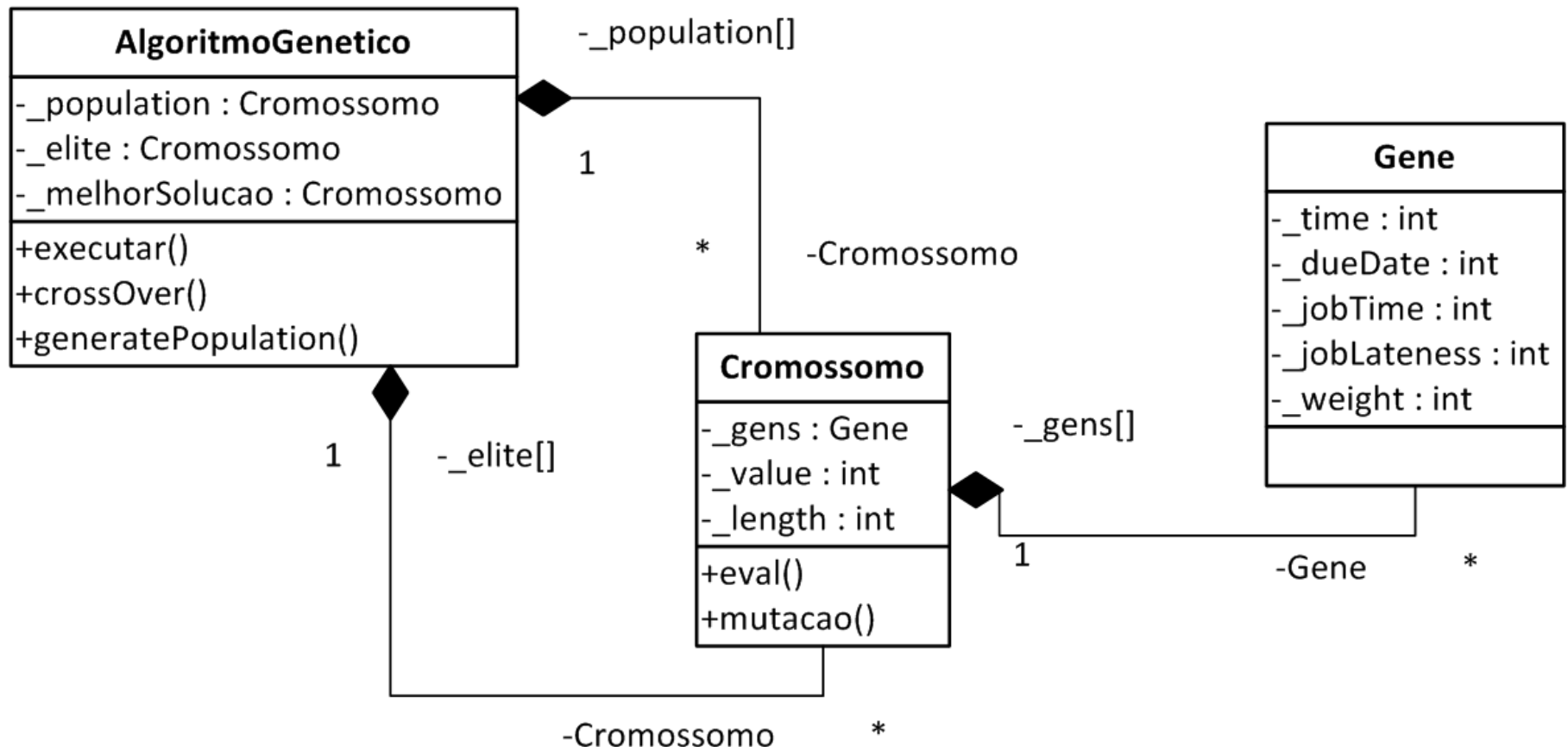
Algoritmos Genéticos

O Algoritmo Genético é uma meta-heurística baseada no processo de evolução natural utilizada para solucionar problemas de otimização com técnicas como a herança de indivíduos, mutação, seleção e recombinação. Soluções candidatas que formam uma população são representadas como cadeia de caracteres denominadas cromossomos. Em cada geração, o valor da função de avaliação do indivíduo é calculado e, baseado nesses valores, os melhores indivíduos são selecionados para gerar a próxima geração.

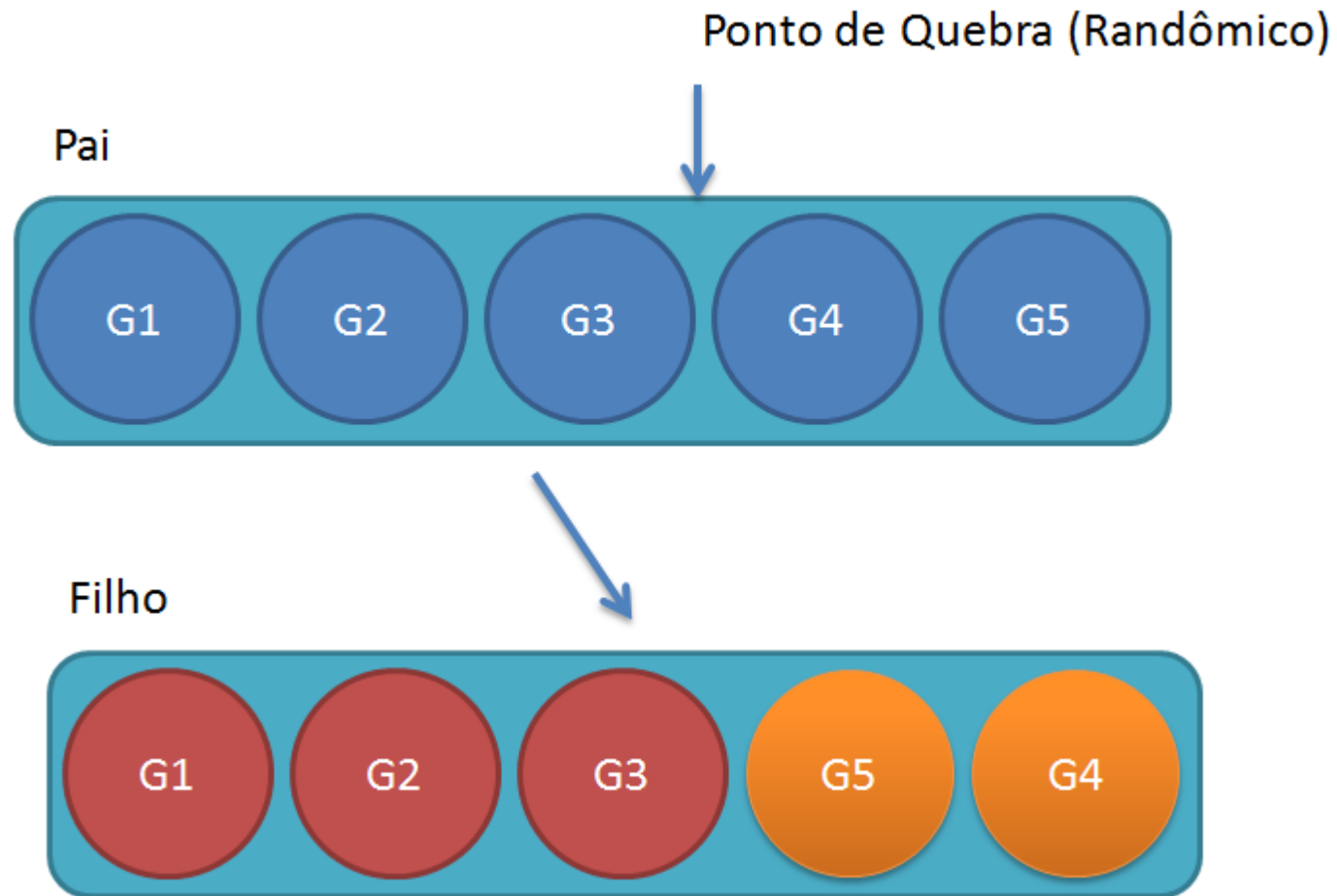
Algoritmos Genéticos

1. Gerar uma população inicial
2. Avaliar a população e parar se critério de parada satisfeito
3. Repetir até indivíduos suficientes
 1. Selecionar pais
 2. Recombinar pais para gerar filhos
 3. Aplicar mutação nos filhos
4. Selecionar nova população entre pais e filhos e voltar para 2.

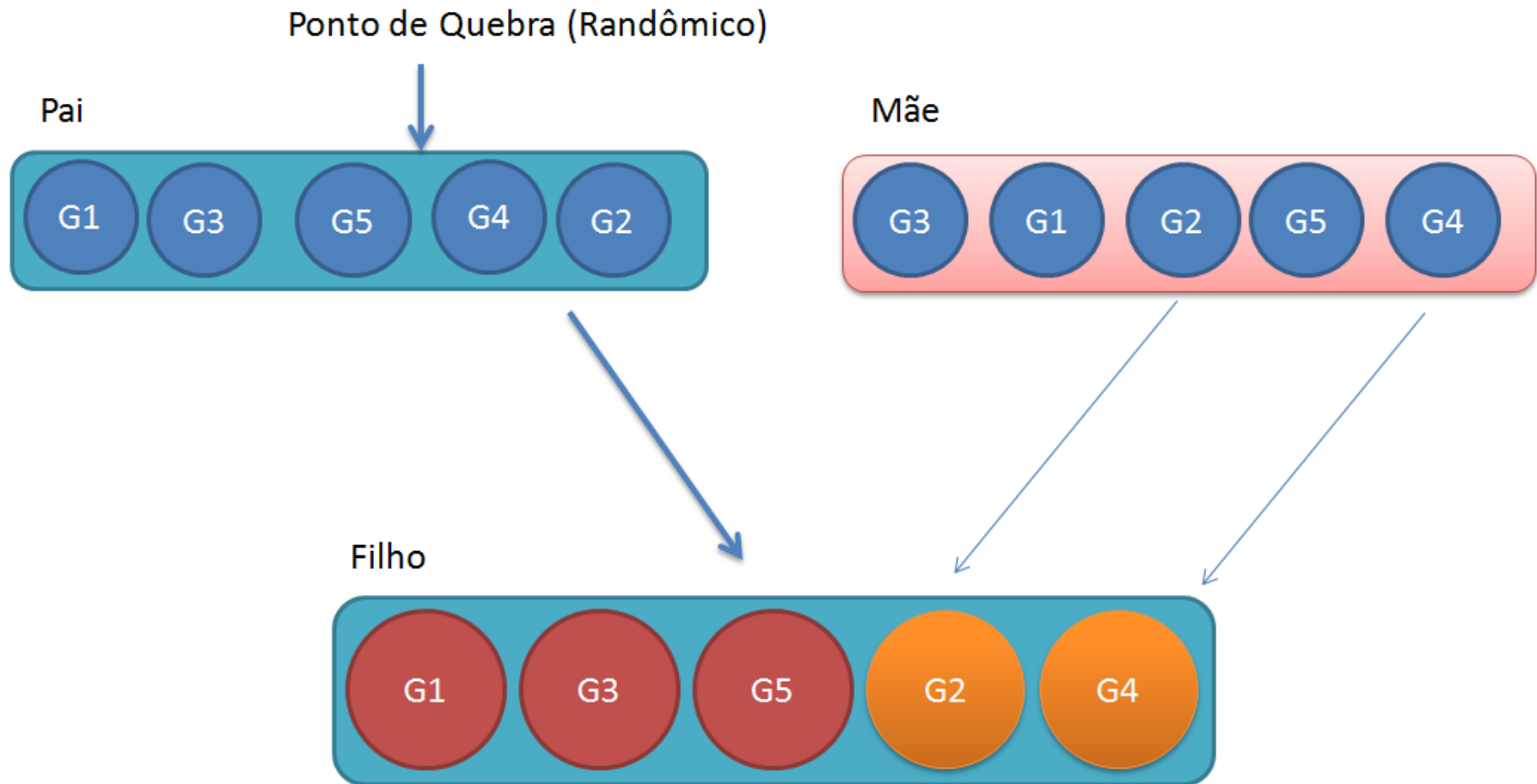
Representação do Problema



Cross-Over (Pai)



Cross-Over (Pai, Mãe)



Mutação (n)

- A mutação acontece sorteando dois genes para trocarem de lugar no cromossomo em questão.
- A mutação ocorre com probabilidade $1/n$.
- Em nosso algoritmo n equivale ao número de indivíduos na população global, ou seja, a probabilidade de ocorrer mutação decai com o crescimento da população.

População e Iterações

1. População Inicial: 25
2. Tamanho da Elite: 25
3. Tamanho máximo da população: 200
4. Tempo mínimo de execução do algoritmo: 1 minuto
5. Critério de parada: tempo de execução

A população inicial é gerada randomicamente com o cromossomo semente que tem as tarefas na ordem em que estão no arquivo de entrada.

A população vai crescendo até chegar ao limite (200). Quando isto acontece, guardamos a população elite e geramos uma nova população.

População e Iterações

A cada iteração geramos quatro filhos de maneiras diferentes.

1. Filho gerado a partir do cross-over(Pai, Mae). Onde o pai é sorteado na população elite e a mãe é sorteada na população global e elite.
2. Filho gerado a partir do cross-over(Pai). Onde o pai é sorteado na população global.
3. Filho gerado a partir da mutação da melhor solução conhecida até o momento.
4. Filho gerado randômicamente.

Testes e Resultados

- Em geral nosso algoritmo obteve em média resultados 3% maior do que o melhor valor conhecido para este problema.
- Em praticamente todas instâncias obtivemos pelo menos uma vez a melhor solução conhecida no tempo de 1 minuto.
- Para cada instância executamos o algoritmo no mínimo 3 vezes.
- O aumento do tempo e/ou população praticamente não melhorou o algoritmo.

Testes e Resultados

Entrada	Tempo (minutos)	Valor Encontrado	Melhor Conhecido	Diferença (%)	Média
WT100/1	1	6077	5988	1,49%	3,34%
WT100/1	1	6066	5988	1,30%	
WT100/1	1	5988	5988	0,00%	
WT100/1	1	5988	5988	0,00%	
WT100/1	1	6294	5988	5,11%	
WT100/1	15	5988	5988	0,00%	
WT100/1	15	6076	5988	1,47%	
WT100/1	15	5988	5988	0,00%	
WT100/1	15	6187	5988	3,32%	
WT100/1	15	6076	5988	1,47%	
WT100/1	30	6076	5988	1,47%	
WT100/1	30	6253	5988	4,43%	
WT100/1	30	6076	5988	1,47%	
WT100/1	30	6076	5988	1,47%	
WT100/1	30	6077	5988	1,49%	
WT50/1	1	2346	2134	9,93%	
WT50/1	1	2184	2134	2,34%	
WT50/1	1	2184	2134	2,34%	
WT50/1	1	2134	2134	0,00%	
WT50/1	1	2345	2134	9,89%	
WT50/1	15	2345	2134	9,89%	
WT50/1	15	2134	2134	0,00%	
WT50/1	15	2184	2134	2,34%	
WT50/1	15	2346	2134	9,93%	
WT50/1	15	2395	2134	12,23%	
WT50/1	30	2184	2134	2,34%	
WT50/1	30	2345	2134	9,89%	
WT50/1	30	2184	2134	2,34%	
WT50/1	30	2184	2134	2,34%	
WT50/1	30	2134	2134	0,00%	

Testes e Resultados

Entrada	Valor Encontrado	Melhor Conhecido	Diferença (%)	Média
WT100/1	6077	5988	1,49%	1,95%
WT100/1	6066	5988	1,30%	
WT100/1	5988	5988	0,00%	
WT100/2	6308	6170	2,24%	
WT100/2	6476	6170	4,96%	
WT100/2	6314	6170	2,33%	
WT100/3	4267	4267	0,00%	
WT100/3	4530	4267	6,16%	
WT100/3	4286	4267	0,45%	
WT100/4	5011	5011	0,00%	
WT100/4	5150	5011	2,77%	
WT100/4	5150	5011	2,77%	
WT50/1	2184	2134	2,34%	
WT50/1	2134	2134	0,00%	
WT50/1	2345	2134	9,89%	
WT50/2	2011	1996	0,75%	
WT50/2	2011	1996	0,75%	
WT50/2	1998	1996	0,10%	
WT50/3	2619	2593	1,00%	
WT50/3	2691	2593	3,78%	
WT50/3	2691	2593	3,78%	
WT50/4	2691	2691	0,00%	
WT50/4	2691	2691	0,00%	
WT50/4	2691	2691	0,00%	
Testes realizados com tempo máximo de 1 minuto.				

Testes e Comparativo

Instância	Melhor valor conhecido	GLPK		Algoritmo Genético				
		Valor obtido	Tempo(min)	Solução Inicial	Solução Final	Desvio SF – SI	Desvio SF – SO	Tempo(min)
WT100/1	5988	58713	240	14251	5988	-137,99%	0,0000%	1
WT100/1	5988	58713	240	14251	6066	-134,93%	1,2859%	1
WT100/2	6170	66092	240	14995	6308	-137,71%	2,1877%	1
WT100/2	6170	66092	240	14995	6314	-137,49%	2,2806%	1
WT100/3	4267	51986	240	9509	4267	-122,85%	0,0000%	1
WT100/3	4267	51986	240	9509	4286	-121,86%	0,4433%	1
WT100/5	5283	43650	240	10852	5283	-105,41%	0,0000%	1
WT100/5	5283	43650	240	10852	5283	-105,41%	0,0000%	1
WT100/7	50972	197790	240	73447	51256	-43,29%	0,5541%	1
WT100/7	50972	197790	240	73447	51293	-43,19%	0,6258%	1
WT100/11	181649	429880	240	214390	181715	-17,98%	0,0363%	1
WT100/11	181649	429880	240	214390	182034	-17,77%	0,2115%	1
WT100/13	178840	473290	240	213154	179539	-18,72%	0,3893%	1
WT100/13	178840	473290	240	213154	180872	-17,85%	1,1234%	1
WT100/17	332804	692306	240	364058	333795	-9,07%	0,2969%	1
WT100/17	332804	692306	240	364058	334020	-8,99%	0,3641%	1
WT100/19	477684	867564	240	507535	477965	-6,19%	0,0588%	1
WT100/19	477684	867564	240	507535	478174	-6,14%	0,1025%	1
WT100/23	539716	1097539	240	545743	539716	-1,12%	0,0000%	1
WT100/23	539716	1097539	240	545743	539720	-1,12%	0,0007%	1

Conclusões

- Problema com complexidade muito alta.
- Meta-heurística
rápida;
resultados satisfatórios;
- Solver
lento;
resultados para tempos "pequenos",
insatisfatório

Referências

- <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/wtinfo.html>
- http://en.wikipedia.org/wiki/Genetic_algorithm
- SOUZA, Marcone J. F., Otimização combinatória
- Ning LIU, M. ABDELRAHMAN e Srini RAMASWAMY., A Genetic Algorithm for Single Machine Total Weighted Tardiness Scheduling Problem,
- KHOWALA, K., KEKA, A., FOWLER, J. – A Comparison of different formulation for the non-preemptive single machine total weighted tardiness scheduling problem.

Muito Obrigado

Perguntas?